**EE422/CS421 Introduction to Robotics Spring 2025**

**Multi Robot Collaboration With Deep Learning**

Group Members

Fadime Yaren Durmuş 200201003

Mustafa Kağan Aytaç 200202026

**Report – Mini Kitchen Simulation**

**Project Description**

This project focuses on designing a simulated environment in which one or more robots can collaboratively prepare a hamburger by interacting with different stations in a grid-based kitchen. The main purpose of the system is to enable these robots to navigate, collect, process, and combine ingredients in the correct order to produce a valid hamburger.

The environment has been carefully designed to support a reinforcement learning (RL) framework. Although the learning component has not yet been implemented due to time constraints, the structure is ready for integration of both traditional Q-learning and more advanced neural network-based reinforcement learning approaches. The final goal is to allow the robots to autonomously learn optimal strategies for completing their tasks, improving over time.

**Environment Development Process**

Initial Design: The kitchen layout was first drafted in an Excel file to visualize station placement and movement areas.

Base Environment (00kitchen.py): Implemented the initial grid with symbolic representations for each station.

Station Integration (01kitchen.py): Added visual representations for key stations:

|  |  |
| --- | --- |
| B = Bread Station  L = Lettuce Station  TO = Tomato Station  C = Cutting Board  P = Pan (Stove)  M = Meat Station  PL = Plate Station  T = Table  D = Delivery  X = Trash  . = Walkable area | A grid of squares with different colored squares  AI-generated content may be incorrect. |

Movement and Robot Control (02kitchen.py): Added a robot agent capable of directional movement (up, down, left, right) along with corresponding animations.

Interface Enhancements (03kitchen.py): Added a sidebar to show current status information of the robot.

Item Handling (04kitchen.py – 06kitchen.py): Enabled robots to hold and drop items.Introduced item states (e.g., raw, chopped, cooked). Implemented interactions with nearby objects using the spacebar. Displayed held and dropped items within the environment.

Cooking and Chopping Logic (07kitchen.py): Robot can chop lettuce/tomato or cook meat at designated stations. Objects on tables are now visible and distinguishable.

Visual Upgrades (08kitchen.py): Replaced text with item-specific PNGs dynamically loaded from the assets folder (e.g., tomato\_raw.png, meat\_cooked.png, plate\_clean.png).

Combination System (09kitchen.py): Implemented logic to combine items placed on or into a plate. Defined over a dozen unique combinations, culminating in the creation of a full hamburger (plate\_burger.png). Once items are combined, they cannot be separated—mimicking real-world cooking constraints.

A screenshot of a game

AI-generated content may be incorrect.

**The Purpose of This Simulation**

The aim of this system is to create a framework where multiple robots can operate in a shared environment to complete structured tasks. In this case, that task is building a hamburger from raw ingredients. Although only one robot is active in the current version, the system is designed to accommodate multiple robots.

The long-term goal is to integrate reinforcement learning, where each robot will learn its own strategy using Q-learning or more complex neural-based methods. This involves creating Q-tables or approximating them using neural networks, enabling robots to optimize their actions over time.

In more advanced stages, multiple agents will learn to cooperate, avoid conflicts, and share the workspace efficiently. Techniques such as Independent Q-Learning, CTDE (Centralized Training, Decentralized Execution), or DQN-based policies can be incorporated to support these capabilities.